

並列計算機の並列処理効率に関する評価の一方法

5 F - 3

富田 和良 北川 一

豊田工業大学工学部

1.はじめに

並列計算機上で並列プログラムを動作させる時、プロセッサをN台にしても、プロセッサ間通信や各プロセッサの処理量の不均衡などのため、並列処理効率が低下し、実行時間は1/Nとはならない。本研究ではプログラムの実行時間を定式化することによって、プログラム特性などのパラメータから並列処理効率を求められるようにし、並列計算機の処理効率に関する評価の一方法を提案する。並列計算機としてはFACOM VPP500を対象とした。VPP500は分散メモリ型のベクトル並列計算機で各プロセッサは1.6GFLOPSの処理性能を持ち、プロセッサ間は通信速度800Mbyte/secのクロスバーネットワークで接続されている。

2.並列処理効率の定式化

プログラムの実行時間を定式化する。N個のプロセッサでプログラムを並列に実行した時の実行時間 $T_{all,N}$ は計算時間とプロセッサ間のデータ転送にかかる時間に分けられる(式(1))。計算時間 T_{calc} は式(2)のように表すことができる。kは各プロセッサでの処理量の不均衡を表すパラメータで、処理量が同じであれば $k=1$ となる。 T_{transf} については並列計算機のプロセッサ間の結合方式や通信速度などのアーキテクチャに依存するので、以下ではVPP500について定式化する。VPP500には大きく分けてbroadcastとspreadmoveによる2種類のデータ転送方法があるため式(3)のように T_{transf} を T_{br} 、 T_{sp} の2つに分け、それぞれについて定式化を行った。ただし、ここでは一般的に示すため転送時間 T_{transf} をデータ転送量などのパ

ラメータを引数に持つ関数として表す(式(3))。

$$T_{all,N} = T_{calc} + T_{transf} \quad (1)$$

$$T_{calc} = T_s + \frac{T_p}{k \cdot N} \quad (2)$$

$$T_{transf} = T_{br} + T_{sp} = f(x_1, x_2, \dots, x_n) \quad (3)$$

$T_{all,N}$: プロセッサ数Nでの実行時間

T_{calc} : 計算時間

T_{transf} : データ転送時間(ただしN=1の時 $T_{transf}=0$)

T_s : T_{calc} のうち並列化不可能な部分

T_p : T_{calc} のうち並列化可能な部分

k : 処理量の不均衡を表す定数
($0 < k \leq 1$ ただしN=1の時, $k=1$)

N : プロセッサ数

T_{br} : broadcastによる転送時間

T_{sp} : spreadmoveによる転送時間

$f(x_1, \dots, x_n)$: データ転送時間 T_{transf} の一般形

x_i : 転送量等の転送時間に関わるパラメータ

次に並列処理効率を表す並列効果Eおよびプロセッサ利用率eを定式化する。並列効果はN台のプロセッサでプログラムを並列実行した時に、実行時間が何倍速くなるかを示す。一方、プロセッサ利用率は並列効果をプロセッサ数Nで割り、プロセッサがどれだけ有効に使われているかを示す。ここで、プログラム特性として、並列化率 $\alpha = T_p / T_{all,1}$ ($0 \leq \alpha \leq 1$)および並列化オーバーヘッド発生率 $\beta = k \cdot N \cdot T_{transf} / T_p$ ($\beta \geq 0$)の2つのパラメータを導入する。 α はプログラムがどれだけ並列化できるかを示し、 β はプログラムを並列化させることによって生じたデータ転送時間とプログラム中の並列計算部分 $T_p / (k \cdot N)$ の比を指している。並列効果、プロセッサ利用率は α 、 β を用いて式(4)、(5)のように表せる。

A method of parallel processing efficiency evaluation on parallel computers.

Kazuyoshi Tomita and Hajime Kitagawa

Faculty of Engineering, Toyota Technological Institute

$$E = \frac{T_{all,1}}{T_{all,N}} = \frac{1}{1 + \left(\frac{1}{k \cdot N} - 1 + \frac{\beta}{k \cdot N} \right) \alpha} \quad (4)$$

$$e = \frac{E}{N} = \frac{1}{N + \left(\frac{1}{k} - N + \frac{\beta}{k} \right) \alpha} \quad (5)$$

3. 定式化の検証

定式化した式(4), (5)が, 実際にプログラムを走らせた時の値にあてはまるかを検証するため, テストプログラムとして有限要素法を用いた構造解析プログラムを作成した. このプログラムでは連立1次方程式を解くルーチンが処理量の大部分を占めている. そこで, VPP500用にこのルーチンを並列化し, プロセッサ数1~10で実行した時の並列効果, プロセッサ利用率を実測し, 式(4), (5)の値と比較する. プロセッサ数を変化させた時の並列効果, プロセッサ利用率と式(4), (5)を計算した結果を図1に示す. 式(4), (5)には構造解析プログラムのプログラム特性 $\alpha = 0.99$, $\beta = (2.37 \cdot \lceil \log_2 N \rceil + 4.18)N \times 10^{-3}$

(ただし, $N=1$ の時, $\beta=0$) を代入し, $k=1$ と仮定した. これは $25 \times 25 \times 5$ にメッシュを切った3次元構造体を解析した時のものである. 式(4), (5)はほぼ実測値にそっている. 実測値がわずかに小さいのは, $k=1$ と理想的な状態を仮定しているためである.

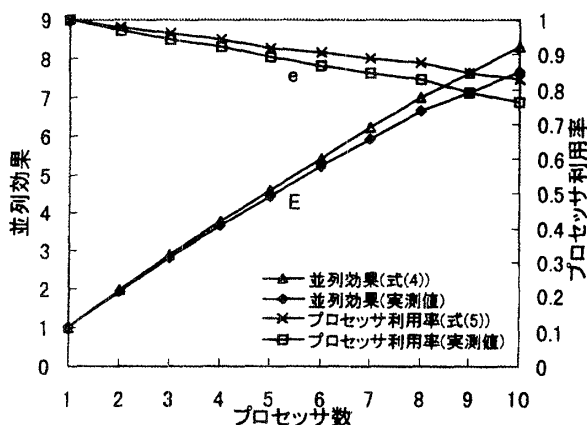


図1 構造解析プログラムの並列処理効率 (VPP500)

4. 並列計算機の性能評価への応用

並列計算機の性能評価への応用方法を示す. プ

ロセッサ数が N の時, プロセッサ利用率が r 以上になるための条件は式(5)より, 式(6)のような不等式になる. プロセッサ数が 10 の時, プロセッサ利用率が 80%以上になる条件は $N=10, r=0.8$ ($k=1$ と仮定)を式(6)に代入すると, プログラム特性 α , β の条件は図2のようになる.

$$\beta \leq \left(\frac{k}{r} - k \cdot N \right) \frac{1}{\alpha} + k \cdot N - 1 \quad (6)$$

次に, ある並列計算機上での並列プログラムを想定して, プログラムの T_s, T_p から α を求め, 並列化アルゴリズムからわかるデータ転送量により β を得る. 得られたプログラム特性 α , β が図2の領域に入っているかどうかで, 想定したプログラムのプロセッサ利用率が 0.8 以上かどうかを判定できる. また α , β を式(4), (5)に代入することによって並列処理効率が計算できる. これを様々なプログラムを想定して試し, その並列計算機でプロセッサ利用率が高くなるのは, どのようなプログラムかという評価が行える.

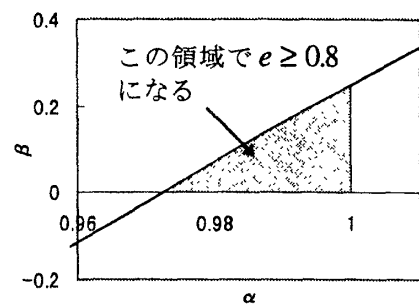


図2 $e \geq 0.8$ となる条件 ($N=10$)

5. まとめ

並列処理効率を定式化し, 並列計算機の並列処理効率に関する評価の一方法を示した. 今後, 実際に VPP500 を対象にして, どのようなプログラムで性能が出るかといった評価を行う. また, パラメータ k の与えかたについても考えたい.

なお, VPP500 については京都大学大型計算機センターのシステムを使用した.

また, この研究について多くのご助言をいただいた豊田工業大学工学部の中川 徹助教授に感謝致します.